/\*selecteer alle familienaam uit actor en voeg daarna een tweede kolom toe die het aantal per naam zal weergeven. Sorteer dan volgens het aantal van hoog naar laag \*/

SELECT actor.last\_name, COUNT(actor.last\_name) as aantal

FROM actor

GROUP BY actor.last\_name

ORDER BY aantal DESC;

1.

SELECT DISTINCT\*

FROM country;

* Zo weet je hoeveel verschillende er effectief zijn

SELECT COUNT(country.country)

FROM country;

2.

SELECT country.country, COUNT(customer.customer\_id) AS aantal

FROM country LEFT JOIN city ON country.country\_id = city.country\_id

LEFT JOIN address ON city.city\_id = address.city\_id

LEFT JOIN customer ON address.address\_id = customer.address\_id

GROUP BY country.country

ORDER BY aantal DESC

LIMIT 5;

3.

SELECT language.name

FROM language

ORDER BY language.name ASC;

4.

SELECT actor.first\_name,actor.last\_name

FROM actor

WHERE actor.last\_name LIKE '%son%'

ORDER BY actor.first\_name ASC;

5.

/\*veel op veel relatie\*/

SELECT film.title,category.name

FROM film LEFT JOIN film\_category ON film.film\_id = film\_category.film\_id

LEFT JOIN category ON film\_category.category\_id = category.category\_id;

6.

SELECT category.name, COUNT(film.title)

FROM category LEFT JOIN film\_category ON category.category\_id = film\_category.category\_id

LEFT JOIN film ON film\_category.film\_id = film.film\_id

GROUP BY category.name;

7.

SELECT actor.first\_name, actor.last\_name, COUNT(film.title)

FROM actor LEFT JOIN film\_actor ON actor.actor\_id = film\_actor.actor\_id

LEFT JOIN film ON film\_actor.film\_id = film.film\_id

GROUP BY actor.first\_name, actor.last\_name;

8.

SELECT film.film\_id,film.title, inventory.film\_id

FROM film LEFT JOIN inventory ON film.film\_id = inventory.film\_id

WHERE inventory.film\_id IS NULL;

* Laatste kolom moet niet getoond worden

SELECT film.film\_id,film.title

FROM film LEFT JOIN inventory ON film.film\_id = inventory.film\_id

WHERE inventory.film\_id IS NULL;

9.

SELECT customer.first\_name, customer.last\_name, customer.customer\_id, rental.customer\_id

FROM customer LEFT JOIN rental ON customer.customer\_id = rental.customer\_id

GROUP BY customer.first\_name, customer.last\_name;

* Onnodige kolommen verbergen

SELECT customer.first\_name, customer.last\_name

FROM customer LEFT JOIN rental ON customer.customer\_id = rental.customer\_id

WHERE rental.rental\_date IS NULL

GROUP BY customer.first\_name, customer.last\_name;

10.

SELECT COUNT(film.title), category.name, language.name

FROM category

LEFT JOIN film\_category ON category.category\_id = film\_category.category\_id

LEFT JOIN film ON film\_category.film\_id = film.film\_id

LEFT JOIN language ON film.language\_id = language.language\_id

WHERE category.name ='Documentary' AND language.name='English'

GROUP BY category.name, language.name;

11.

SELECT\*

FROM actor

WHERE first\_name = 'SCARLETT';

12.

SELECT\*

FROM actor

WHERE last\_name = 'Johansson';

13.

SELECT COUNT(DISTINCT actor.last\_name)

FROM actor;

14.

SELECT actor.last\_name, COUNT(actor.last\_name) as aantal

FROM actor

GROUP BY actor.last\_name

HAVING aantal = 1;

15.

SELECT actor.last\_name, COUNT(actor.last\_name) as aantal

FROM actor

GROUP BY actor.last\_name

HAVING aantal > 1;

OF

SELECT actor.last\_name

FROM actor

GROUP BY actor.last\_name

HAVING COUNT(actor.last\_name) > 1;

* Hier zie je geen kolom met aantallen

16.

SELECT actor.actor\_id, actor.first\_name, actor.last\_name, COUNT(actor.actor\_id) AS aantal

FROM actor LEFT JOIN film\_actor ON actor.actor\_id = film\_actor.actor\_id

LEFT JOIN film ON film\_actor.film\_id = film.film\_id

GROUP BY actor.actor\_id, actor.first\_name, actor.last\_name

ORDER BY aantal DESC

LIMIT 1;

SELECT actor.actor\_id, actor.first\_name, actor.last\_name

FROM actor LEFT JOIN film\_actor ON actor.actor\_id = film\_actor.actor\_id

LEFT JOIN film ON film\_actor.film\_id = film.film\_id

GROUP BY actor.actor\_id, actor.first\_name, actor.last\_name

HAVING COUNT(actor.actor\_id)

ORDER BY COUNT(actor.actor\_id) DESC

LIMIT 1;

17.

SELECT film.title, inventory.store\_id, rental.rental\_date, rental.return\_date

FROM rental LEFT JOIN inventory ON rental.inventory\_id = inventory.inventory\_id

LEFT JOIN film ON inventory.film\_id = film.film\_id

WHERE film.title='Academy Dinosaur';

* Huidig rental aanbod

SELECT film.title, inventory.store\_id, rental.rental\_date, rental.return\_date, inventory.inventory\_id

FROM rental LEFT JOIN inventory ON rental.inventory\_id = inventory.inventory\_id

LEFT JOIN film ON inventory.film\_id = film.film\_id

WHERE film.title='Academy Dinosaur' AND inventory.store\_id='1' AND rental.return\_date IS NOT NULL;

OF

SELECT film.title, inventory.store\_id,inventory.inventory\_id, count(inventory.inventory\_id) as aantal

FROM rental LEFT JOIN inventory ON rental.inventory\_id = inventory.inventory\_id

LEFT JOIN film ON inventory.film\_id = film.film\_id

WHERE film.title='Academy Dinosaur' AND inventory.store\_id='1' AND rental.return\_date IS NOT NULL

GROUP BY inventory.inventory\_id;

18. WACHTEN

19. TO DO

SELECT film.title, inventory.store\_id, rental.rental\_date, rental.return\_date, inventory.inventory\_id, film.rental\_duration

FROM rental LEFT JOIN inventory ON rental.inventory\_id = inventory.inventory\_id

LEFT JOIN film ON inventory.film\_id = film.film\_id

WHERE film.title='Academy Dinosaur' AND rental.return\_date IS NULL;

20.

21.

22.

23.

24.

25.

26.

27.

28.

29.

30.

31.

32.

33.

34.

SELECT staff.first\_name, payment.amount,payment.payment\_date

FROM staff

LEFT JOIN payment ON staff.staff\_id = payment.staff\_id

WHERE payment.payment\_date LIKE '2005-08%';

OF

SELECT CONCAT(staff.first\_name, ' ', staff.last\_name) as naam, SUM(payment.amount) AS som

FROM staff

LEFT JOIN payment ON staff.staff\_id = payment.staff\_id

WHERE payment.payment\_date BETWEEN '2005-08-01' AND '2005-08-31'

GROUP BY naam;

35.

36.

37.

38.

39.

40.

41.

42.

43. WACHTEN